

## SOFTWARE PARTITION OF MIDI SYNTHESIZER FOR HOST/DSP (OMAP) ARCHITECTURE

### Background of the Invention

#### 1. Field of the Invention

This invention relates generally to third generation (3G) wireless phones and more particularly to a technique for implementing MIDI functionality without use of an ASIC in a handset having a multimedia processor such as OMAP.

#### 2. Description of the Prior Art

Third generation (3G) wireless phones will have the processing power to run a wide variety of applications, including multimedia. MIDI sequencing and synthesis is one application required by handset manufacturers and carriers. A stand alone, fixed function ASIC that handles MIDI sequencing and synthesis is generally used as a solution. These 3G handsets may include a multimedia processor such as OMAP that defines the operations, maintenance and application protocol. The stand alone, fixed function ASICs add to the overall system cost and are also problematic since the integrated MIDI code is difficult or impossible to update dynamically.

In view of the foregoing, a need exists for a scheme to implement MIDI functionality on OMAP, eliminating the need for the ASIC.

**Summary of the Invention**

The present invention is directed to a system and method of implementing a software solution for 3G phones that use OMAP and require MIDI synthesis.

5 Specifically, 3G phones that use OMAP and require MIDI synthesis can replace a fixed function ASIC solution with an OMAP software solution, yielding a reduced system cost. The system and method implements optimal task partitioning between a general purpose processor (GPP) and a digital signal processor (DSP).

10 According to one embodiment, a MIDI synthesizer is implemented in OMAP using flash memory to allow dynamic updates of sample sets.

According to another embodiment, a MIDI synthesizer is implemented in OMAP using flash memory to allow news instruments to be added to the synthesizer.

15 According to yet another embodiment, a MIDI synthesizer is implemented in OMAP using flash memory to allow MIDI code to be dynamically updated from a network.

According to still another embodiment, a MIDI synthesizer is implemented in OMAP using flash memory to optimize PCB space and minimize costs associated with 3G handsets.

20 According to still another embodiment, a MIDI synthesizer is implemented in OMAP using flash memory to optimize use of DSP memory by only loading required sample sets from flash for any given MIDI file.

**Brief Description of the Drawings**

Other aspects and features of the present invention and many of the attendant advantages of the present invention will be readily appreciated as the same become better understood by reference to the following detailed description when considered in connection with the accompanying drawings in which like reference numerals designate like parts throughout the figures thereof and wherein:

Figure 1 is a block diagram illustrating a system and method for implementing a MIDI software partition for OMAP according to one embodiment of the present invention; and

Figure 2 is flowchart illustrating GPP and DSP threads for the system and method depicted in Figure 1.

While the above-identified drawing figures set forth alternative embodiments, other embodiments of the present invention are also contemplated, as noted in the discussion. In all cases, this disclosure presents illustrated embodiments of the present invention by way of representation and not limitation. Numerous other modifications and embodiments can be devised by those skilled in the art which fall within the scope and spirit of the principles of this invention.

### **Detailed Description of the Preferred Embodiments**

Third generation (3G) wireless phones will have the processing power to run a wide variety of applications, including multimedia. MIDI sequencing and synthesis is one application that is required by handset manufacturers and carriers. A stand alone, fixed function ASIC typically handles the MIDI sequencing and synthesis. The present inventor realized handsets that include a multimedia processor such as OMAP can implement MIDI functionality on OMAP, eliminating the need for the ASIC, and achieve a significant cost reduction.

Software implementations for heterogeneous multi-processor configurations, such as OMAP, require careful design considerations to achieve optimized performance in terms of power and memory requirements. Specifically, the way software is partitioned across different processors is critical. A system and method of partitioning MIDI sequencing and synthesis software for OMAP to achieve optimal performance according to one embodiment is described herein below with reference to Figures 1 and 2.

Figure 1 is a high level block diagram illustrating a system and method for implementing a MIDI software partition for OMAP according to one embodiment of the present invention. An ARM9 general purpose processor (GPP) 12 handles the task of reading and parsing MIDI files stored in a flash memory 18 and sending the appropriate synthesizer commands to the digital signal processor (DSP) 14. Since the GPP 12 is byte addressable, it is more efficient in parsing a MIDI stream of byte granularity. The DSP 14 handles the media intensive task of audio synthesis and rendering the audio to the digital-to-analog converter (DAC) 16. The flash memory 18 architecture, dual management and control (MAC) and DSP peripherals 20 for driving the DAC 16 make the DSP 14 more efficient for this task. The DSP peripherals 20 include a C55x chipset commercially available from Texas Instruments Incorporated of Dallas, Texas.

Figure 2 is flowchart 100 illustrating GPP 12 and DSP 14 threads 102, 104 for the system and method 10 depicted in Figure 1. The interaction of threads 102, 104 is initiated when the GPP 12 first opens a MIDI bit stream as shown in block 106. The GPP 12 will perform an initial examination of the MIDI bit stream to determine what sample sets must be loaded to DSP 14 memory. The GPP 12 will then load and instantiate (make

an instance of code and data that can be executed) the DSP code as shown in blocks 108 and 110 respectively, initialize the sample set memory (that may reside in either shared memory, or DSP memory) and signal the DSP 14 to start running the DSP synthesizer as shown in block 112. Subsequently, the GPP 12 will parse the MIDI data into synthesis packets as shown in blocks 114 and 116, which are then transferred to the DSP 14. The DSP 14 takes the time stamped MIDI commands, synthesizes them and renders the audio to the DAC 16 as depicted in block 118. The foregoing process continues until the MIDI bit stream is completely read as shown in blocks 120 and 122, or until the user terminates the application. At that point, the GPP 12 signals the DSP 14 to stop running the synthesis thread, de-allocate memory used, and finally terminate the thread as shown in blocks 124 and 126.

In view of the above, it can be seen the present invention presents a significant advancement in the art of 3G handsets that include a multimedia processor such as OMAP to implement MIDI functionality on OMAP, eliminating the need for an ASIC to achieve a significant cost reduction. Further, this invention has been described in considerable detail in order to provide those skilled in the data communication art with the information needed to apply the novel principles and to construct and use such specialized components as are required. In view of the foregoing descriptions, it should be apparent that the present invention represents a significant departure from the prior art in construction and operation. However, while particular embodiments of the present invention have been described herein in detail, it is to be understood that various alterations, modifications and substitutions can be made therein without departing in any way from the spirit and scope of the present invention, as defined in the claims which follow.